

main.c

```
Init()
→ init all functions
Test LEDs
Test Motors

Loop forever
{
  Light_Intensity==0?
  (is at light source)
  → drive away fast

  good_bearing?
  → braitenberg

  else?
  → random walk
}
```

ISR(TimerB0,TimerB0_ISR)

- ADC_Grab() (@ ADC10.c)
 - Start ADC conversion
- send IR

ISR(TimerA1,TimerA1_ISR)

- receive IR

ISR(ADC10,ADC_ISR)

Triggered when ADC conversion is ready.

- cyclic change channels
PD1/2/3, Batt, Food0/1
- read ADC value

./behav/braitenberg.c

- Implement Braitenberg behavior based on PD1/2/3 Readings.
- Control motors

motor.c

ISR(WATCHDOG,WatchdogISR)

- Random walking (if enabled)
- Generate motor PWM

ISR(TimerA0,TimerA0_ISR)

- should never be triggered
Not used

ISR(TimerB1,TimerB1_ISR)

- should never be triggered
Not used